Practical Work 03 (Queues and Stacks)

Objectives: Manipulation of Queues and Stacks and mastering with them.

Exercise 1

Implement a queue using a singly linked list in C. Include the following operations:

- enqueue(int data): Add an element to the rear of the queue.
- dequeue(): Remove an element from the front of the queue.
- isEmpty(): Check if the queue is empty.
- display(): Display all elements in the queue.

Exercise 2

Implement a priority queue where elements are dequeued based on their priority (e.g., higher priority elements are dequeued first). Use an array or linked list to implement this.

Exercise 3

Simulate a printer queue. Use a queue to manage the order of requests and process them one by one. (use modules isempty,isfull, enqueue, dequeue and printprocess which execute all print operation using the bellows module).

Exercise 4

Write a C program to implement a stack using an array. Include the following operations:

- push(): Add an element to the top of the stack.
- pop(): Remove and return the top element from the stack.
- peek(): Return the top element without removing it.
- isEmpty(): Check if the stack is empty.
- isFull(): Check if the stack is full (if using a fixed-size array).

Exercice 5

Write a C program that uses a stack to reverse a string. Push each character of the string onto the stack, then pop them off to get the reversed string.

Exercise 6

Write a C program to check if a given expression has balanced parentheses using a stack. The program should handle different types of brackets: (), {}, [].

Exercise 7

Write a C program to implement a stack using a singly linked list. Include the standard stack operations (push, pop, peek, isEmpty).

Exercise 8

Write a C program to find the next greater element for each element in an array using a stack. The next greater element for an element x is the first greater element on the right side of x in the array. If no such element exists, consider it as -1.

Exercise 9

Write a C program to solve the stock span problem using a stack. The stock span problem is a financial problem where we have a series of daily price quotes for a stock and we need to calculate the span of the stock's price for all days. The span of the stock's price on a given day is defined as the maximum number of consecutive days (starting from today and going backward) for which the stock price was less than or equal to today's price.

Exercise 10

Write a C program to evaluate a given arithmetic expression that includes parentheses, addition, subtraction, multiplication, and division. Use stacks to handle the parentheses and operator precedence.